A faster-converging algorithm for image segmentation with a modified Chan-Vese model

Rick Chartrand and Valentina Staneva

***Abstract*—We propose an algorithm for segmentation of grayscale images. Our algorithm computes a solution to the convex, unconstrained minimization problem proposed by T. Chan, S. Esedoḡlu, and M. Nikolova in [1], which is closely related to the Chan-Vese level set algorithm for the Mumford-Shah segmentation model. Up to now this problem has been solved with a gradient descent method. Our approach is a quasi-Newton method based on the lagged diffusivity algorithm [2] for minimizing the total-variation functional for image denoising [3]. Our results show that our algorithm requires a much smaller number of iterations and less time to converge than gradient descent, and is able to segment noisy images correctly.**

***Keywords*—image segmentation, Mumford-Shah segmentation, Chan-Vese model, quasi-Newton method**

I. Introduction

THE Mumford-Shah energy model is among the best-known models for image segmentation. It finds a piecewise smooth approximation u of a grayscale image f defined on the region D by minimizing the following functional:

$$F(u,\Gamma) = \mathcal{H}^1(\Gamma) + \lambda \int_D (u(x)-f(x))^2\,dx + \mu \int_{D\setminus\Gamma} |\nabla u(x)|^2\,dx. \quad (1)$$

The first term is the 1-dimensional Hausdorff measure of the set of shape boundaries Γ, and accounts for the smoothness of the boundaries. The second term is the L_2-distance between u and f, and ensures the result is close to the original image f. The third term is responsible for the minimization of the L_2-norm of the gradient of u, which smooths the image outside the set of shape boundaries Γ.

A simpler version is obtained by restricting u to be piecewise constant, and taking on only two values, as considered by T. Chan and L. Vese [4]:

$$F(c_1,c_2,\Sigma) = \int_D |\nabla \chi_\Sigma(x)|\,dx + \lambda \int_D \left[\chi_\Sigma(x)(c_1-f(x))^2 + (1-\chi_\Sigma(x))(c_2-f(x))^2\right] dx. \quad (2)$$

When $\Sigma \subset D$ has a rectifiable boundary, the first term is the length of $\partial\Sigma$. The second term corresponds to the second term of (1), with u having the value c_1 on Σ and c_2 on $D\setminus\Sigma$. Note that the third term of (1) is zero in this context.

Rick Chartrand is with the Mathematical Modeling and Analysis Group, Los Alamos National Laboratory, Los Alamos, email: rickc@lanl.gov

Valentina Staneva is with the Applied Mathematics & Statistics Department, Johns Hopkins University, Baltimore, email: staneva@ams.jhu.edu

Even with this simplification, it is difficult to minimize (2): the minimization is done over three unknown variables, with Σ ranging over subsets of the plane. The Chan-Vese active contour model [4] is a reformulation that can be handled by standard level-set methods [5]. In this model the boundary Γ is represented as the 0-level set of a function ϕ. Then the Mumford-Shah energy is rewritten in the following way:

$$F_{CV}(c_1,c_2,\phi) = \int_D |\nabla H_\epsilon(\phi(x))|\,dx + \lambda \int_D H_\epsilon(\phi(x))(c_1 - H_\epsilon(\phi(x)))^2\,dx + \lambda \int_D (1-H_\epsilon(\phi(x)))(c_2-f(x))^2\,dx, \quad (3)$$

where H_ϵ is a regularization of the Heaviside function H, and c_1 and c_2 are the average intensity values of the two regions into which the image is split during the segmentation.

The functional (3) is nonconvex, and the solution can depend on the initial ϕ that is used. T. Chan, S. Esedoḡlu, and M. Nikolova propose a convex alternative in [6]. For fixed c_1, c_2, a minimizer of $F(c_1,c_2,\cdot)$ in (2) can be obtained by solving the convex, constrained minimization problem

$$\min_u F(u) = \int_D |\nabla u(x)|\,dx + \lambda \int_D \left[(c_1-f(x))^2 - (c_2-f(x))^2\right] u(x)\,dx, \quad (4)$$

subject to $0 \le u(x) \le 1$, and then thresholding the solution: $\Sigma = \{x \in D : u(x) \ge \mu\}$, for any μ outside a set of measure zero. (Note how substituting $u = \chi_\Sigma$ gives a quantity differing from (2) by a constant depending only on c_2 and f.) One can then update c_1 and c_2, and iterate the process: $c_1 = \int_\Sigma f(x)\,dx/|\Sigma|$ and $c_2 = \int_{D\setminus\Sigma} f(x)\,dx/|D\setminus\Sigma|$. They then further show that (4) can be replaced by an equivalent, convex, unconstrained problem:

$$\min_u F(u) = \int_D |\nabla u(x)|\,dx + \lambda \int_D \left[(c_1-f(x))^2 - (c_2-f(x))^2\right] u(x)\,dx + \alpha \int_D \max\{0, 2|u(x)-1/2|-1\}\,dx, \quad (5)$$

provided $\alpha > \frac{\lambda}{2}\|(c_1-f)^2 - (c_2-f)^2\|_{L^\infty(D)}$. The third term penalizes u having values outside $[0,1]$. The result is a much simpler optimization problem, and is similar to standard models for total variation regularization. Since it is convex and unconstrained, it can be solved by a gradient descent method,

which is what is used in [6]. In their numerical results, they obtain near-binary solutions, allowing image segmentation without thresholding. The advantage of the gradient descent method is that it can be easily implemented. However, it requires many iterations to converge. In our work we propose a faster algorithm for the same problem.

II. The proposed algorithm

To solve the segmentation problem we adapt the lagged diffusivity method [2] for minimizing the ROF functional. The similarity between the two models is that they both require minimization of the total variation of the image plus a data fidelity term. This relation to the denoising algorithm allows us to apply to the Chan-Esedoğlu-Nikolova model a technique similar to the one used by C. Vogel and M. Oman in [2] to linearize the problem and quickly find the solution.

We simplify our notation by letting

$$\nu(\xi) = \max\{0, 2|\xi - 1/2| - 1\}, \tag{6}$$

and

$$s(x) = (c_1 - f(x))^2 - (c_2 - f(x))^2. \tag{7}$$

After the substitution the functional in (5) becomes:

$$F(u) = \int_D |\nabla u(x)|\, dx + \int_D \big(\lambda s(x)u(x) + \alpha\nu(u(x))\big)\, dx. \tag{8}$$

This functional is non-differentiable, which we resolve by approximating the first term with a differentiable function, while choosing a representation for ν' for our numerical computations. We approximate $|\nabla u(x)|$ by $|\nabla u(x)|_\epsilon$, where $|v|_\epsilon := \sqrt{|v|^2 + \epsilon^2}$ for a small $\epsilon > 0$. For ν' we simply make choices from the subgradient, and use:

$$\nu'(\xi) = \begin{cases} -2, & \xi < 0, \\ 0, & 0 \le \xi \le 1, \\ 2, & \xi > 1. \end{cases} \tag{9}$$

We obtain an Euler-Lagrange equation of (5) by setting the (approximate) derivative of (8) equal to zero:

$$F'(u) = -\nabla \cdot \frac{\nabla u}{|\nabla u|_\epsilon} + \lambda s + \alpha\nu' \circ u = 0. \tag{10}$$

The equation in (10) is highly nonlinear in terms of u. As done in [2] for the ROF functional, we solve the equation iteratively by approximating the nonlinear terms with their values from the previous iteration:

$$-\nabla \cdot \frac{\nabla u_{n+1}}{|\nabla u_n|_\epsilon} + \lambda s + \alpha\nu' \circ u_n = 0. \tag{11}$$

If we denote $L(u)v = -\nabla \cdot \frac{\nabla v}{|\nabla u|_\epsilon}$, the equation becomes:

$$L(u_n)u_{n+1} + \lambda s + \alpha\nu' \circ u_n = 0. \tag{12}$$

Under the assumption that u_{n+1} would not differ much from u_n we approximate (12) as:

$$L(u_n)u_{n+1} + u_{n+1} - u_n + \lambda s + \alpha\nu' \circ u_n = 0. \tag{13}$$

This can also be thought of as adding and subtracting u in (10) and then lagging one of the terms. Now we can solve for u_{n+1}:

$$u_{n+1} = (L(u_n) + I)^{-1}(u_n - \lambda s - \alpha\nu' \circ u_n). \tag{14}$$

The solution can be done without the step in (13), but this way we ensure that the matrix we are inverting in (14) is nonsingular, as $L(u)$ is positive semidefinite. We can also rewrite the iterative scheme in the following way:

$$\begin{aligned} u_{n+1} &= u_n - (L(u_n) + I)^{-1}(L(u_n)u_n + \lambda s + \alpha\nu' \circ u_n) \\ &= u_n - (L(u_n) + I)^{-1}F'(u_n). \end{aligned} \tag{15}$$

The iteration (15) can be regarded as a quasi-Newton method for the solution of (10), with $L(u)+I$ being an approximation of the second derivative $F''(u) = L(u) + L'(u)u + \nu'' \circ u$. The reason we do not approach the problem with a real Newton's method, which is known to have a locally quadratic rate of convergence, is that $L'(u)u$ is expensive to compute and store. The approximation $L(u) + I$ is appropriate for the algorithm: it is a positive definite matrix, therefore, our step direction $u_{n+1} - u_n$ is guaranteed to be a descent direction [7]. Finally, we include a timestep, which in practice we hold constant:

$$u_{n+1} = u_n - \Delta t(L(u_n) + I)^{-1}F'(u_n). \tag{16}$$

Our numerical experiments show that the algorithm successfully segments images, and converges faster than the gradient descent method.

III. Results

We test the algorithm on the 256×256 cameraman image displayed in Figure 1. Gaussian noise with variance of 0.001 was added to the image, which causes simple thresholding techniques to fail in segmenting it. We would like the result to be close to a binary image. As a starting point we use the noisy image. We first implement the gradient descent method. We choose our parameters to achieve a satisfactory segmentation: $\lambda = 1000$, $\alpha = 1000$, $\gamma = 0.0001$. The largest working timestep is 0.000001, and the number of iterations needed to obtain the image in Figure 2(a) is 5000, which requires 830 s of CPU time (in MATLAB). We also implement our quasi-Newton method (16). The appropriate timestep for this case is $\Delta t = 0.1$. After 200 iterations and 260 s of CPU time the algorithm obtains the solution shown in Figure 2(b). We display how the value of the functional changes with each iteration for both algorithms in Figures 2(c) and 2(d). The gradient descent method decreases the value slowly, and even after 5000 iterations has not completely converged. On the other hand, the quasi-Newton method makes it drop quickly to a very small value after only 50 iterations, and then gradually continues to improve the solution. In the end, the quasi-Newton method's minimum is still lower than the minimum found by the gradient descent method. Visually, the solution of the quasi-Newton method does not differ much from the one achieved through the gradient descent method. Simple thresholding of this result can convert it into a binary image of cameraman on a white background.

We also test the algorithm on a different example in which we try to segment a simple object from a dark background,

Fig. 1. The cameraman image with some Gaussian noise added to it: we would like to obtain a binary segmentation in which the cameraman is separated from the background.

but the image has been corrupted by Gaussian noise with variance equal to 0.03 (Figures 3(a), 3(b)). It is easy for the human eye to distinguish the object, since we possess some intuition for the shape of the football. However, for a computer algorithm it is difficult to correctly reconstruct the boundaries since no prior knowledge is assumed. We attempt to segment the image with the popular k-means algorithm which partitions the intensity values into k clusters. Since this method does not assume any connectivity of regions, it is very sensitive to noise and it fails to select the shape (Figure 3(c)). The proposed algorithm, on the other hand, is trying to preserve the geometric information in the image. The noise increases the amount of variation in the image, so in order to come to a piecewise constant solution, the regularization parameter λ needs to be smaller. This increases the number of necessary iterations: more work needs to be done to obtain segmentation. Since we want to be able to segment images efficiently even when noise is present, we change the starting point from the noisy image to a crude segmentation of it. For example, we use the result from the k-means algorithm. We could also first start the segmentation with a larger λ. Finally we obtain our final result displayed in Figure 3(d) in only 40 iterations (as opposed to 200 when starting with the noisy image). We note that despite the noise the image is close to binary, and does not require additional thresholding.

REFERENCES

[1] M. Nikolova, "Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers," *SIAM J. Numer. Anal.*, vol. 40, pp. 965–994, 2002.

[2] C. R. Vogel and M. E. Oman, "Iterative methods for total variation denoising," *SIAM J. Sci. Comput.*, vol. 17, no. 1, pp. 227–238, 1996.

[3] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, pp. 259–268, 1992.

[4] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, pp. 266–277, 2001.

[5] S. Osher and J. A. Sethian, "Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.*, vol. 79, pp. 12–49, 1988.

[6] T. F. Chan, S. Esedoḡlu, and M. Nikolova, "Algorithms for finding global minimizers of image segmentation and denoising models," *SIAM J. Appl. Math.*, vol. 66, pp. 1632–1648, 2006.

[7] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge: Cambridge University Press, 2004.

PLACE PHOTO HERE

Rick Chartrand received his Ph.D. in Mathematics in 1999 from UC Berkeley, where he studied functional analysis. He now works as an applied mathematician at Los Alamos National Laboratory. His research interests include compressive sensing, image processing, and the high-dimensional geometry and analysis of images and data.

PLACE PHOTO HERE

Valentina Staneva received her B.Sc. from Concord University in 2006. She is now a Ph.D. student in the Applied Mathematics department at John Hopkins University.

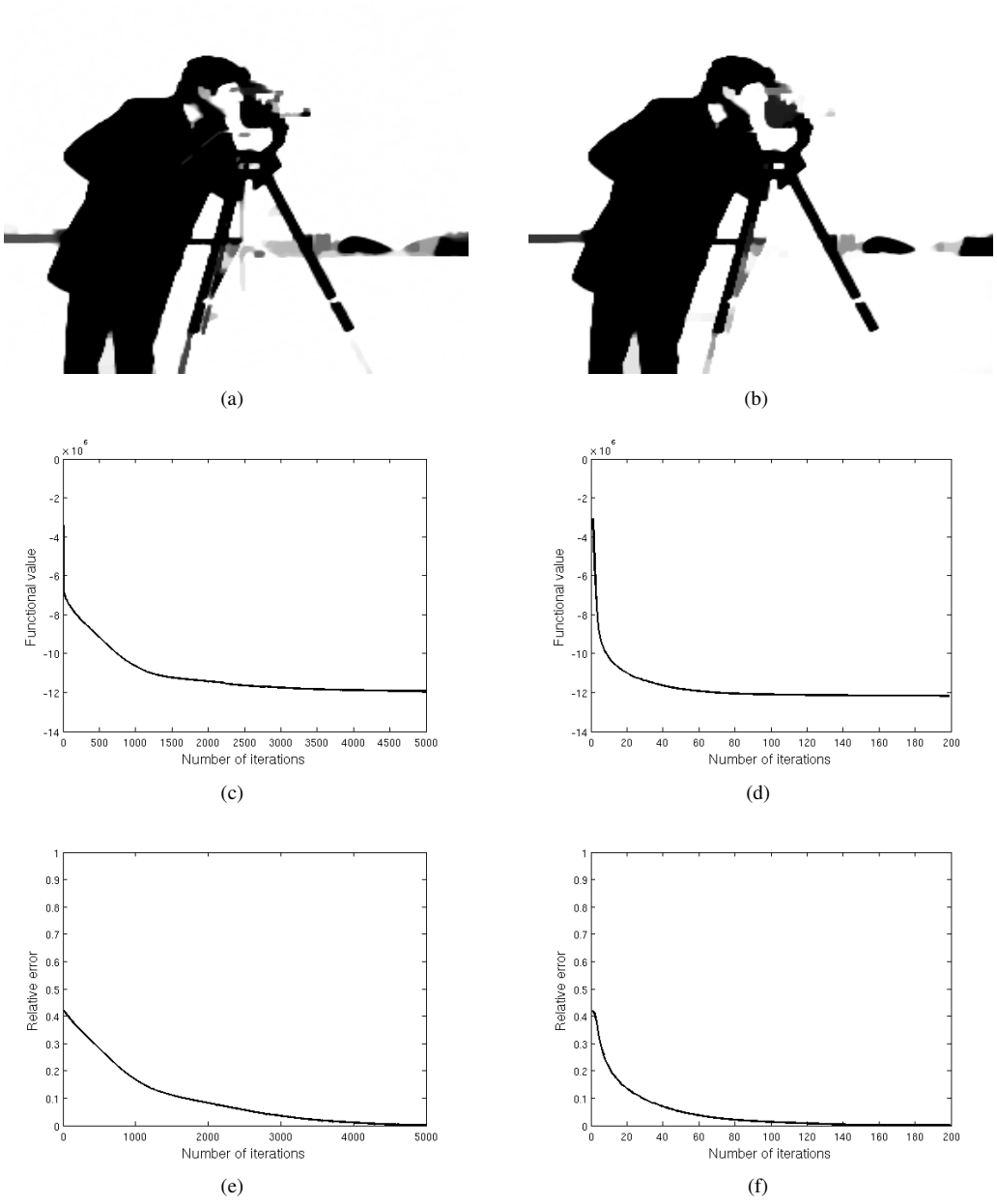
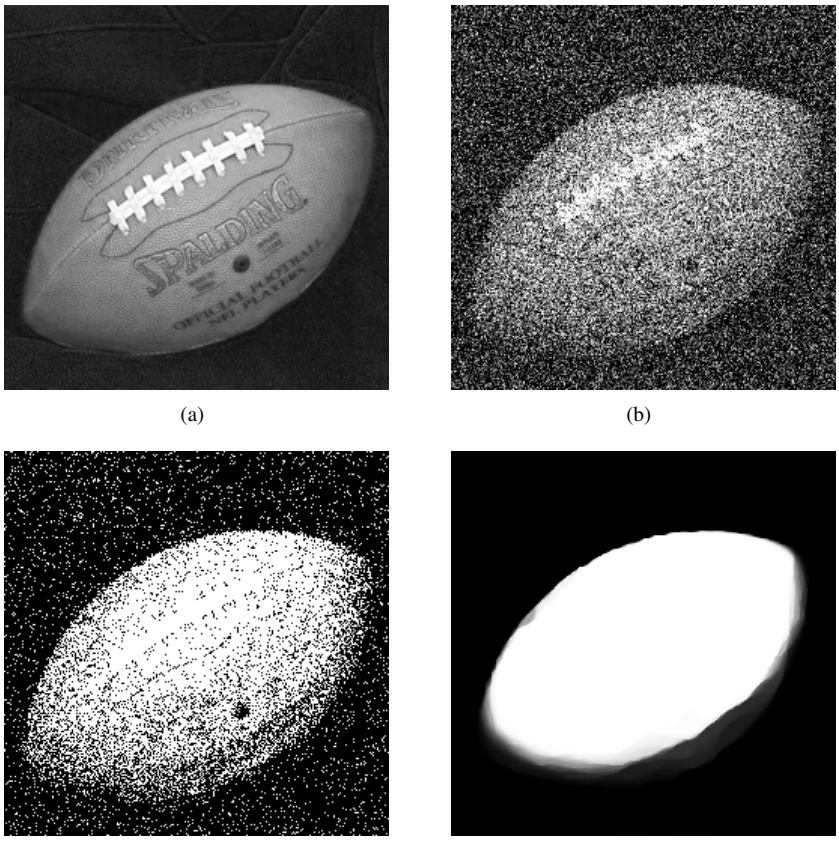


Fig. 2. Segmentation results of the cameraman image obtained with: (a) a gradient descent method, and (b) our quasi-Newton method; (c) the decrease of the functional value with each iteration of the gradient descent method; (d) the decrease of the functional value with each iteration of the quasi-Newton method (note that $s(x)$ in (8) can be negative, but is bounded below); (e) the relative error ($\|u_n - u_{opt}\|/\|u_{opt}\|$) for (c); (f) the relative error for (d).



(a) (b)

(c) (d)

Fig. 3. (a) An image of a football: a simple segmentation problem. (b) A significant amount of noise is added to the image, which makes simple segmentation techniques fail. (c) Unsuccessful segmentation of the image with the *k*-means algorithm. (d) Segmentation achieved through the proposed algorithm; as an initial guess we used the poor k-means segmentation in order to reduce the number of iterations.